BOLT BERANEK AND NEWMAN INC

CONSULTING · DEVELOPMENT · RESEARCH

Report No. 2541 April 1973

INTERFACE MESSAGE PROCESSORS FOR THE ARPA COMPUTER NETWORK

QUARTERLY TECHNICAL REPORT NO. 1
1 January 1973 to 31 March 1973

Principal Investigator: Mr. Frank E. Heart
Telephone (617) 491-1850, Ext. 470

Sponsored by
Advanced Research Projects Agency
ARPA Order No. 2351

Contract No. F08606-73-C-0027
Effective Date: 1 January 1973
Expiration Date: 31 December 1973
Contract Amount: $2,147,071

Title of Work: IMP Program

Submitted to:

IMP Program Manager
Range Measurements Lab.
Building 981
Patrick Air Force Base
Cocoa Beach, Florida 32925

Report No. 2541 Bolt Beranek and Newman Inc.

INTERFACE MESSAGE PROCESSORS FOR THE ARPA COMPUTER NETWORK

QUARTERLY TECHNICAL REPORT NO. 1

1 January 1973 to 31 March 1973

Submitted to:

IMP Program Manager
Range Measurements Lab.
Building 981
Patrick Air Force Base
Cocoa Beach, Florida 32925

This research was supported by the Advanced Research Projects Agency of the Department of Defense under Contract No. F08606-73-C-0027.

TABLE OF CONTENTS

	Page
1. OVERVIEW	1
1.1 IMP/TIP Memory Retrofit Program	3
1.2 The TIP's Network Status Facility	5
1.3 Network Traffic	7
2. NETWORK RELIABILITY	11
2.1 IMP Program Changes	12
2.2 TIP Program Changes	16
3. HSMIMP DEVELOPMENT	19

1. OVERVIEW

This Quarterly Technical Report, Number 1, describes aspects of our work on the ARPA Computer Network during the first quarter of 1973.

During this quarter we installed one 316 IMP, at Lawrence Berkeley Laboratory, and one TIP, at the Range Measurements Laboratory. Also, the 316 IMP which had previously been in operation at Tinker Air Force Base was removed from the network during the quarter. At the end of the quarter the network included 35 operational IMPs and TIPs plus an experimental TIP at BBN. The first quarter saw the installation of the first "Very Distant" Host, between Speech Communications Research Laboratory, Inc. and the UCSB IMP.

A major activity during the first quarter has been making IMPs less sensitive to hardware failures in themselves or their neighbors. The motivation for, and conclusions of, this effort are described in Section 2.

Work on the High Speed Modular IMP (HSMIMP) continued through the first quarter. Section 3 provides a survey and progress report on the HSMIMP development.

We completed the implementation of a Network Control Program, and some TELNET capabilities, for our PDP-1 during the first quarter. At one time we intended to make Network Control Center (NCC) traffic summary data available to the network community via TELNET on the PDP-1, but after a recent expression of ARPA preference have decided to store this data on a TENEX machine instead.

We continue to study the problems of subnetwork routing. During the past quarter these studies have concentrated on two topics: possible improvements in the speed and efficiency of routing propagation, and area routing. On the former topic, we have been particularly interested in developing routing propagation methods which allow for a variety of line speeds and allow for bursts of traffic while at all times providing unambiguous routing information even in the face of severe network transients. On the latter topic we have been particularly interested in developing techniques which allow hierarchies of areas while minimizing the possibility a node can be cut off from other nodes; this seems to require dynamic configuration of areas. Additionally, we considered routing over a broadcast channel and routing based on available capacity rather than delay in view of our new ideas for area routing and propagation of routing.

During the first quarter our satellite communications effort centered around three areas: study of the long-term placement of Satellite IMPs; continued simulation and analysis of the Reservation-ALOHA and other related satellite protocols; and study, analysis, and simulation of algorithms for slotting a broadcast channel.

The SIMP placement study resulted in the recommendation that SIMPs be placed in the satellite ground stations. Continued simulation and analysis of Reservation-ALOHA led to better understanding of its weaknesses (a tendency toward instability is one) and its strengths (it is generally comparable to the Interleaved Reservations system of Roberts, with both achieving almost complete channel utilization in the face of reasonable delays). The analysis and simulation of slotting algorithms has led to the conclusion that slotting is relatively easy for a 50 Kbs channel and possible for a megabit rate channel.

We continue to be heavily involved in network Host protocol development. During the past quarter we helped to organize two protocol workshops; one of these specified a major revision of the TELNET Protocol and the other refined the File Transfer Protocol. In addition to our technical contributions at these workshops, BBN is producing the documentation of the new protocols. We have also been involved in an exchange of ideas about Host protocol with the International Network Working Group.

Two papers describing aspects of our work on the network project were presented during the first quarter. One, entitled *A System for Broadcast Communication: Reservation-ALOHA*, was presented at the 1973 Hawaii System Science Conference. The second, *Terminal Access to the ARPA Network: Experience and Improvements*, which describes directions of TIP development, was presented at COMPCON 73 (the seventh annual IEEE Computer Society International Conference).

1.1 IMP/TIP Memory Retrofit Program

Since the beginning of 1973 we have been involved in a program of retrofitting additional core memory to all IMPs and TIPs in the field. This change increases the core size of IMPs to 16K words (from 12K) and the core size of TIPs to 28K (from 20K); the TIP magnetic tape option continues to require an additional 4K. The IMP core expansion was necessitated by the continuing growth of the network and the expanded algorithms (such as area routing) which will eventually be needed to deal with that growth. Originally unforeseen options (such as the "Very Distant" Host option) also made an increase in memory size desirable. In the TIP, an increasing diversity of terminal types, a desire for larger terminal buffers, and especially a desire for additional features all require more core storage. By the end of the quarter, we had completed the retrofit effort for all but five machines.

Unfortunately, the retrofit program has had some negative effects on network reliability, especially during February and March. First, of course, is the fact that for many machines the retrofit has removed the machine from use for several hours, or in a few cases, more than one day. This always breaks one network path, and usually results in a singly-connected "stub" of one or more other sites. At such times the network is unusually vulnerable to single IMP or line failures elsewhere. We can, however, frequently replace a "missing" machine by simply wiring the output signal from one modem to the input of a second modem through "patch cords" at the *customer* side of the modems. Since each modem is running a data clock, and the clocks are independent, this approach causes bits to occasionally be dropped or picked as the clocks drift or jitter relative to each other. (In fact, if the two clocks were running at opposite ends of their allowed frequency range, bits would be dropped/picked at a very rapid rate.) Because of this, the "composite" line looks much more noisy to the IMPs at each end than either of the common-carrier's circuits (typically a 5% to 10% error rate). Nevertheless, this approach does allow us to maintain network connectivity during extended IMP outages, since the normal IMP-IMP checksum and retransmission scheme insures message integrity through the noise.

A much more serious problem has been the sporadic failure of new memory modules hours, days, or even weeks after their installation and testing. It did not appear economically feasible, at the beginning of the retrofit effort, to set up a memory module "burn in" facility at BBN; thus the retrofit components are not subjected to the same type of pre-installation testing under our control as is used for entire machines. Aside from relatively straightforward machine failures (i.e., IMP solidly down) which are attributable to retrofit activity, we have experienced a few more difficult problems involving the loss of only a few bits of memory; the results of these are discussed in Section 2.

1.2 The TIP's Network Status Facility

As previously mentioned in our Quarterly Technical Report No. 15,* a TIP command has been available for some time which automatically connects the user to a Network Status Facility (formerly NEWS). The Network Status Facility resided in the TENEX system at BBN and, at the beginning of the first quarter, provided three types of service:

- A "Netnews" system which allowed the TIP system programmers to communicate to users.
- A "Gripe" system which allowed users to communicate to the programmers.
- A "Host Status" system which reported which service Hosts in the network were up and available.

During the past quarter we have been working with the TENEX group at BBN in their development of a system called the Resource Sharing Executive (RSEXEC) as this system pertains to the goal of offering improved service to TIP users. Currently, the RSEXEC is supported only on TENEX computers, although expansion to other PDP-10's is expected soon, and expansion to other types of machines may come eventually.

The overall design objective of the RSEXEC is to allow a user to view all of the participating machines as a single resource; the cooperation among the machines required to permit this view should be accomplished through the ARPA Network in ways which are invisible to the user. This concept can be extended to support a TIP user by providing him with a (TIP) command which causes connection with the "nearest" (e.g., the quickest to respond) machine which supports the RSEXEC. Once connected to this system,

*See footnote, page 9.

the user can be offered the full processing power provided by the big system as though that processing were being provided directly from the TIP. In fact, if the TIP were to automatically connect all users to the RSEXEC, the naive user would be able to view its command language as the TIP's command language.

By the end of the first quarter the Network Status Facility had been moved from its somewhat constrained support by BBN's TENEX to the more general support of the RSEXEC. The three previously-available systems are supported as well as the following new additions:

- Describe - A mechanism which provides descriptions of the other available commands.
- Link - A mechanism to allow on-line communication among two or more terminal users connected to RSEXEC.
- Sndmsg (send message) - A general purpose "mail" distribution facility.
- Trminf (terminal finder) - A mechanism which tells a TIP user which TIP Multi-line Controller port he is connected to (this is especially helpful to TIP users connected via dial-in facilities).
- A variety of facilities for text editing (e.g., "delete character" and "delete line") and terminal control (e.g., "full duplex" and "set attention character").

We believe that continued expansion of the cooperation between TIPs and the RSEXEC will allow the TIP to concentrate on the terminal-handling functions which it must provide, while at the same time providing the computer features and facilities which users find convenient and pleasant. This seems directly in line

with the original goal for TIPs, namely to provide easy and inexpensive access for terminal users to extensive computational power.

1.3 Network Traffic

We have been collecting and summarizing network traffic at the Network Control Center (NCC) for somewhat over 18 months, and it seems appropriate to present some of the data at this time. The NCC has been primarily concerned with two kinds of traffic data: first, the total number of packets sent into the network by each Host each month, and second, the percentage of total bandwidth used on each communication circuit. The Host packet output is further subdivided into *internode* traffic (source and destination at different IMPs/TIPs) and *intranode* traffic (source and destination at the same IMP/TIP).

Figure 1 shows an 18 month plot of the average daily internode Host traffic. The average, of course, includes weekends and holidays as well as business days. The data points fit quite well to a straight line (on a log scale) drawn between the first and last data points; thus the average daily network traffic appears to be increasing at a steady exponential rate. The traffic is increasing by an order of magnitude approximately every 13 months, or doubling approximately every four months. (During this 18-month period the number of nodes has merely doubled, from 18 to 36.) During the month of March 1973, the network carried almost two million packets per day; if the average packet length is assumed to be 250 bits this amounts to about .5 billion bits per day. It is also interesting to note that almost all messages are single-packet messages.

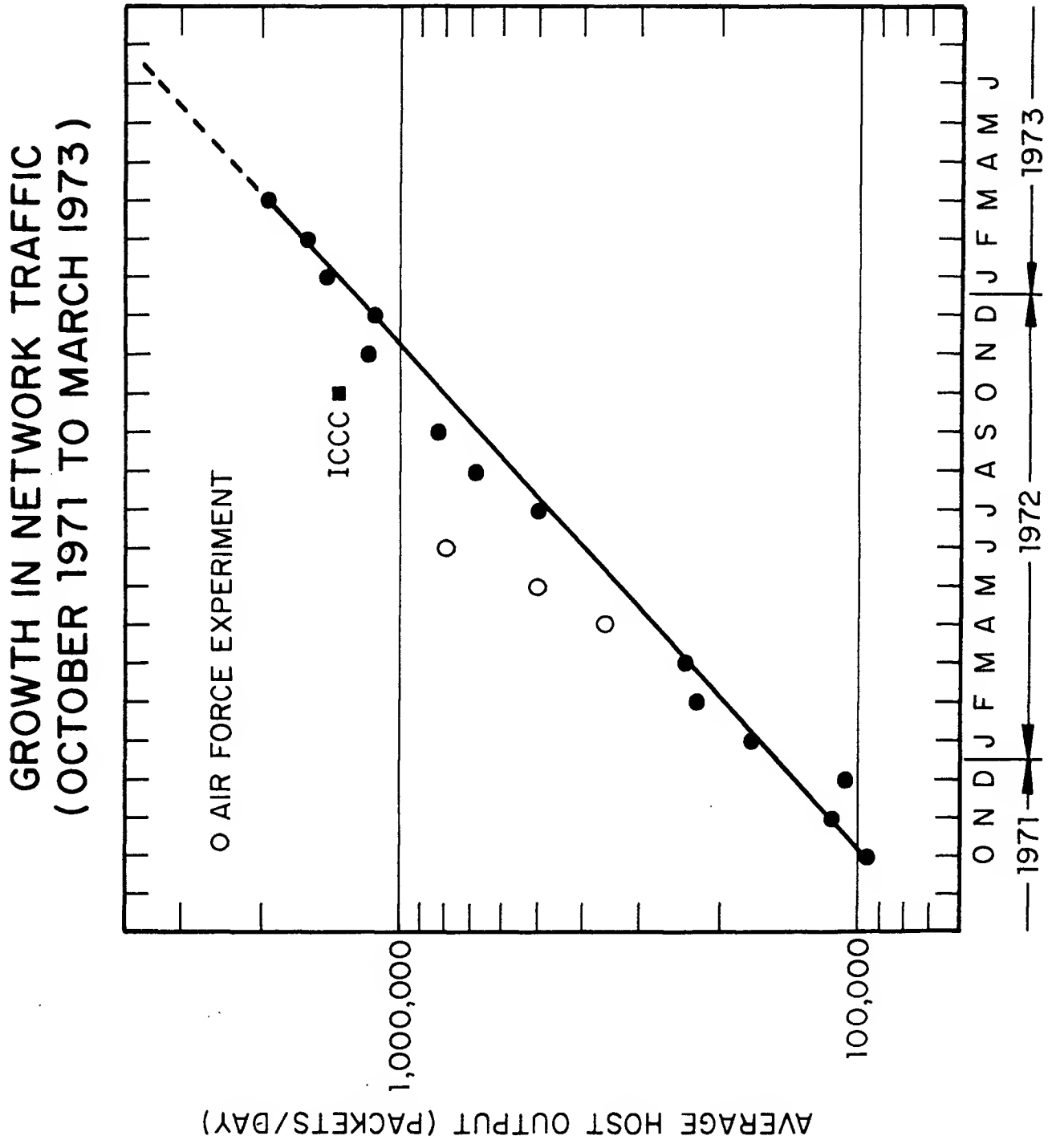
GROWTH IN NETWORK TRAFFIC
(OCTOBER 1971 TO MARCH 1973)
O AIR FORCE EXPERIMENT
ICCC
AVERAGE HOST OUTPUT (PACKETS/DAY)
1,000,000
100,000
O N D J F M A M J J A S O N D J F M A M J
1971
1972
1973

FIGURE 1

There are four data points which do not lie on the curve but can be easily explained. During the last half of April and all of May and June 1972, the U.S. Air Force was conducting network experiments involving large data transfers between Tinker and McClellan Air Force Bases. These experiments were terminated on the last day of June. In October 1972 a TIP was temporarily installed at the International Conference on Computer Communication, as reported in our Quarterly Technical Report No. 16.* During the three days of this conference, and the three preceding days, this TIP was observed to generate over 50,000 messages per hour, with a corresponding traffic load at the Hosts from which it was obtaining service.

It is important for the NCC to observe the communication circuit bandwidth used as a predictor of saturation of particular circuits. There has, of course, been an increase in bandwidth used which corresponds to the increase in internode traffic, but saturation does not appear to be imminent. The useful bandwidth of a circuit is a function of overall circuit bandwidth, IMP-IMP overhead, and packet length. For the 50 Kb circuits which comprise almost all of the network, the useful bandwidth is about 37 Kb when the average message length is 500 bits and about 22 Kb when the average message length is 125 bits (in each of the above cases, RFNM's are counted as messages with length zero). If it is assumed that the traffic is half RFNM's, then the average message length is one half the average Host message length. Table 1 summarizes the average line and busiest line data collected during the month of March, for two assumptions about average Host message length.

*Previous Quarterly Technical Reports were written under Contract No. DAHC-15-69-C-0179.

TABLE 1

	Average Messages/Day	Line Utilization 1000 bits/message	Line Utilization 250 bits/message
Average line	244,649	3.83%	1.61%
Busiest line	519,982	8.14%	3.42%

The average numbers, of course, do not give a true picture of possible saturation; for this one must also know the peak-to-average traffic ratio. During the past 18-month period we have computed an approximation to this ratio several times and found the approximation to lie between three-to-one and four-to-one. (For these approximations we found, in order, the busiest line, its busiest day, and the busiest hour in that day, and compared with the average hourly traffic on that line.) Thus it would appear that the peak traffic that any line is handling cannot be much greater than about one-third of its capacity, and is more likely to be about 10% of capacity.

2. NETWORK RELIABILITY

One of the major activities of the technical staff during the first quarter was a more direct and intensified attempt to improve network reliability. As described in Section 1.1, the memory retrofit program which began this quarter contributed to sporadic failures in several machines which had network-wide effects. In addition, we had already experienced prolonged hardware failure syndromes unrelated to the retrofit program at several sites in the network, particularly in the Washington, D.C. area. The normal procedures of calling in Honeywell and working with Honeywell field engineers had not cleared up several of these persistent failures, and it was felt that an escalation of BBN involvement was needed to identify the exact causes of the problems. Therefore, during much of February and March there were one or more members of the staff at various sites in the network where hardware problems were suspected. The first thing we found out was that the operational IMP program did not give enough diagnostic information about failures when they occurred, and that the available test programs did not detect errors frequently enough to justify their use. That is, the errors were appearing at rather low frequency, from once every few hours to once every few days. Therefore, we decided to try to make the operational IMP program run when it could, and report more information about detected hardware errors, rather than keep the failing IMPs off the network for days at a time. At the same time, we decided to make the TIP programs more resilient, so that TIP users would be less affected by those failures which did occur.

April 1973

2.1 IMP Program Changes

Modifications to the IMP program had two independent goals: we wanted to make the software less vulnerable to hardware failures, and we wanted the software to isolate the failures and report them to the NCC. The technique we chose to use was generating a software checksum on all packets as they are sent out over a line, and verifying the checksum on all packets received on a line. We suspected that the hardware failures in the Washington area were happening between IMPs, that is, the packets were correct before they were sent. The failures could be in the transfer from memory to output modem interface, or in the output interface, or in the input modem interface at the other machine, or in the transfer from the input modem interface to memory. Thus, a memory-to-memory software checksum should be able to detect these errors. (Note that this is in addition to the hardware checksum from output modem interface to input modem interface.) On March 13, a new version of the IMP program was released with software checksum code. It uses a simple checksum, the sum of all the words in a packet minus its length, on all inter-IMP transactions, routing messages, data packets, RFNMs, etc. The reduction in effective processor bandwidth for the 516 IMP is from twenty 50-kilobit lines to fifteen 50-kilobit lines. Any packet found to have an incorrect checksum is discarded, a copy of the data is sent to the NCC, and the previous IMP retransmits the packet.

A partial list of the hardware problems that were uncovered by software checksums, and subsequently fixed, includes:

- One modem interface at the Aberdeen IMP dropped several bits from several successive words in transferring data into memory.

- One modem interface at the Belvoir IMP picked one or two bits in a single word in transferring data into memory.
- One modem interface at the ETAC TIP dropped the first word in transferring data out of memory.

There were other problems that were not detected by the software checksum, such as dropped interrupts. This set of problems may be explained by poor engineering of the high-speed DMC on 316 IMPs. All of the machines cited above are 316 IMPs with 3 modem interfaces, and they are the only such machines in the network. The third interface is in a separate drawer and the total bus length seems to be too long for the driving electronics in the original design. We are presently investigating various ways to fix these problems.

This first experience proved the value of a software checksum on all inter-IMP transmissions. We have decided to extend the checksum to detect intra-IMP failures as well, at the same time cutting the cost of the checksum code by a factor of two. We can obtain an end-to-end software checksum on packets, without any time gaps, as follows:

- A checksum is computed at the source IMP for each packet as it is received from the source Host.
- The checksum is verified at each intermediate IMP as it is received over the circuit from the previous IMP.
- If the checksum is in error, the packet is discarded, and the previous IMP retransmits the packet when it does not receive an acknowledgment.

- The previous IMP verifies the checksum before it retransmits the packet. If it finds an error, it has detected an intra-IMP failure, and the packet is lost. If not, then the first transmission was in error due to an inter-IMP failure, and the previous IMP holds a good copy of the packet.
- After the packet has successfully traversed several intermediate IMPs, it arrives at the destination IMP. The checksum is verified just before the packet is sent to the Host.

This technique provides a checksum from the source IMP to the destination IMP, plus fault isolation to a particular IMP if there is an error. It is half as costly as the first inter-IMP checksum, because only one checksum calculation is performed for each hop, except when there is a retransmission. We intend to install this checksum algorithm in the network shortly.

At the same time that we were discovering inter-IMP failures with the software checksum packets, we began to notice a different kind of problem with intra-IMP failures. In these cases we were primarily faced with memory problems, and they often affected the IMP program itself, rather than the packets flowing through the IMP. Our first attack on this problem was to build a PDP-1 program to verify the running IMP and TIP programs at a site against the correct core images held at the PDP-1. The program interrogates the IMP with DDT messages, and prints out a list of discrepancies. Using this program, we found memory failures at the Lincoln IMP and elsewhere. We soon discovered that even this step was not enough to guarantee network reliability. The core verifier approach assumes that some bits in a word were changed in a core write or a core re-write after a read, or by a runaway program.

But it also assumes that the main body of the IMP program is intact so that it can respond to DDT commands. We had two experiences at this time which illustrated that memory failures can be much more catastrophic.

Specifically, there were two occasions when the routing code in an IMP was incorrect due to a memory failure. Routing messages are particularly important for network reliability, because if one IMP is sending out incorrect routing information (broadcasting that it is the best route to all destinations, for instance) the rest of the network will suffer quickly and on a very large scale. The problems we had were due to single broken instructions in the part of the IMP program that builds the routing message. As a result, the routing messages from that IMP were random data, and the neighboring IMPs interpreted these messages as routing update information. When this happened, traffic flow through the network was completely disrupted and no useful work could be done until the failed IMP was halted.

This kind of problem can happen in three ways:

- The routing message is changed in transmission. The inter-IMP checksum should catch this. The bad routing messages we saw in the network had good checksums.
- The routing message is changed as it is constructed, say by a memory or processor failure, or before it is transmitted. This is what we termed an intra-IMP failure.
- The routing program is incorrect for hardware or software reasons.

We intend to solve the last two kinds of problems by extending the concept of software checksums. The routing program can build a software checksum for the routing message as it builds the message, just as if it came from a Host. Modem output can then always verify the checksum on routing messages before transmitting them. This scheme should detect all intra-IMP failures. Finally, the routing code can calculate a checksum of its own instructions before executing them, to detect any changes in the code. If a discrepancy is found, the program will request a reload immediately. These changes to improve the reliability of routing will also be installed in the network shortly.

2.2 TIP Program Changes

The hardware difficulties which we began to experience during the first quarter had two effects on Host-to-Host communication. First, the intermittant modem interface failures, of the type seen at Belvoir, Aberdeen, and ETAC, meant that messages were occasionally lost by the network. This loss is reported to the transmitting Host by the "Incomplete Transmission" message generated by the source IMP; the Host must then decide whether to retransmit or to take some other action. Second, the higher than normal incidence of machine failures meant that the network sometimes "partitioned" so that there was no path between the two communicating Hosts. (It should be noted that, contrary to the original design, one current site is connected to the network by only a single path; other similar connections are planned. For any such sites, any failure along the single path will be seen as a partition.) Since a TIP acts as a Host for its users, its resilience when these types of failures occur has a major effect on user satisfaction.

Prior to this quarter the TIP program "aborted" the user's connection if it received an Incomplete Transmission indication from the IMP program. During the quarter the TIP program (and the programs of several other Hosts) was changed to retransmit messages for which the Incomplete Transmission indication was returned. On the other hand, it has not seemed reasonable to continue attempting to transmit when the program receives a "Destination Unreachable" indication, since this could arise either from a network partition or from a failure at the destination site. The interactive user is, of course, free to try again manually.

A different situation pertains to tape transfers involving TIPs with the magnetic tape option. In these cases, the user would like to start the process and then ignore it until the transfer is finished. Network partitions, even if infrequent, are too frequent when tape transfers many hours in length are in progress. Therefore, we made a significant modification to the TIP magnetic tape option to include a sequencing mechanism in the tape transfer protocol which permits automatic recovery and transmission continuation after most kinds of network transients. With this mechanism in effect, and assuming a tape is mounted at the "other end", the complete transfer of a tape is possible with a single command given at either end. If the connection goes dead in mid-transfer, the TIP magnetic tape software will attempt to reopen the connection until successful and then continue the transfer from where it was left off. In addition to modifying the TIP magnetic tape option as specified above, we also modified the TENEX program which is able to communicate with the TIP magnetic tape option so that it remained compatible.

3. HSMIMP DEVELOPMENT

The development of the High Speed Modular IMP (HSMIMP) continued to be a major activity during the first quarter. The broad HSMIMP system design has congealed and a paper which describes the system has been prepared, submitted and accepted for presentation at the National Computer Conference in June. Work on both the hardware and software has been progressing and, with the exception of certain possible perturbations discussed below, we appear to be pretty well on schedule. The system design has held up excellently in general, and our conviction that this is an exciting new way to produce reliable computer systems of varying size and power has deepened.

Our working relationship with Lockheed has gone quite well. We remain convinced that the choice of the SUE architecture was correct although the fact that the machine is a relatively new one has created a number of difficulties. We have had to press for some developments that needed speeding up and some fixes to bugs or undesirable features of the Lockheed systems.

With regard to our own hardware development activity, many of the special cards have been built and debugged in prototype form. These include a real time clock, the pseudo-interrupt flag card, the bus coupler cards, a modem interface, a Host interface, a DMA (memory-device block transfer) card and some other minor cards. The pseudo-interrupt card has been produced in final printed circuit form and the bus coupler cards are presently being converted to this form as well. We are also experimenting with an alternate manufacturing approach, known as

"multi-wire", for some of the other cards. This might be an economical alternative to printed circuit multi-layer cards for small quantities. We have rented a semi-automatic wire wrap machine which has speeded and cheapened production of wire wrapped boards.

We have built several test programs and run them on a three bus system, and are presently producing additional bus couplers so that a four bus system can be assembled and tested. The bus couplers have been resolved into three card types. A BCP card sits at the processor end of couplers to either memory or I/O busses. A BCM card sits at the memory and I/O ends of processor couplers, as well as at the memory end of I/O to memory couplers. A third card type, BCI, similar to the BCP but simpler, lies at the I/O bus end of I/O to memory couplers. The BCP's and BCM's, which form the great preponderance of the coupler cards in a system, are debugged, are being built in small quantities in wire-wrap form, and are being laid out for printed circuit manufacture. The BCI is in prototype test.

Our attention has been focused primarily on the central system problems associated with getting a reasonable sized multi-bus prototype working. We have therefore deferred working on producing more esoteric interfaces such as high speed modem and Host interfaces, satellite modem interfaces, etc. Nonetheless, these units have received a good deal of thought. The high speed (1.5 megabit) modem interface, for example, is compatible in most ways with the standard 50 kilobit modem interface — in part because a number of sophistications such as elastic buffering have been made part of the standard interface. The 306 modem requires different drivers and receivers but, except for speed, is functionally similar to the 303. We therefore do not foresee any particular problems in constructing the faster interface.

Another area, that of terminal handling, is benefitting from work on the related RJE effort discussed in Section 1.1 of our Quarterly Technical Report No. 16. Although terminal development is not specifically scheduled for this year, we have nonetheless discussed this matter at some length in order to be sure that our overall system design permits incorporation of terminal handling in a reasonable way.

We have begun building a set of test programs for testing systems and sub-systems at all levels. This work will certainly continue throughout the rest of the year. Inasmuch as we will be producing a number of the special cards ourselves, at least for the present, we must produce test set-ups and programs to check out these cards. In addition, of course, flexible programs must be prepared to test a wide variety of system configurations. Our test programming has just about been keeping pace with our prototype designs. We now have some test programs which exercise systems of a few busses in size.

Now that the designs of many of the cards are completed and their general characteristics (size, power and cooling requirements, etc.) better understood, we are taking a second look at a number of system-wide physical issues concerning power, cooling, and mechanical layout and modularity. We have, until recently, been planning to build our prototype into two six-foot cabinets. However, a recent decision to enlarge the prototype to test out a dual I/O bus configuration, coupled with the need for some extra working space to experiment with varied configurations, caused the system to overflow the second cabinet. In addition, we are facing some problems with cooling and inter-bus and inter-cabinet cabling. We are therefore presently investigating another approach which would package and cool units in a more modular

fashion. This review has led to a reconsideration of the bus coupling cables as well. In our prototype bus couplers we have been using two 2-inch wide flat cables to interconnect the ends of bus couplers. We are currently considering ways to utilize fewer and more manageable cables.

A further area of concern has been the development of faster and more compact memories for the SUE. Because of the system quantization to single bus size, because memories tend to fill up processor and memory busses, and because the memories are not quite fast enough to be shared locally between the processors on a processor bus, the system design could benefit greatly from a faster and physically smaller memory. In addition, the lack of parity on the SUE memories, although they apparently perform well, is a source of concern to us. We have therefore explored, and are continuing to explore, the possibility of obtaining an IC memory system through a number of vendors, including Lockheed. We have been somewhat discouraged by our findings to date; we have been unable to find a vendor willing to supply a compatible memory of appropriate size and speed for a reasonable price. This situation, however, seems likely to change as 4K memory chips become commercially available.

Coding for the new machine has started, and a DDT has been constructed. The conversion of the IMP algorithms from one machine to another represents a large but straightforward task. The fact that the new machine is a multiprocessor leads to some new technical concepts.

Our basic control passing mechanism, as described in Section 4 of our Quarterly Technical Report No. 15, is to break the program up into 300-microsecond "strips", and to resample the Pseudo-Interrupt Device (PID) at the end of each strip to determine what

strip to execute next. At this writing, this control mechanism seems easy to code for, efficient, and easier to debug than the interrupt philosophy of the 516 IMPs.

Our basic locking mechanism is the read and clear instruction, implemented in the bus coupler. It too seems to be working out well, although we have not tried using it in a multi-processor system yet.

Coding of the basic store/forward path has progressed to the point where some of its crucial properties can be calculated by counting instructions. Most of our original assumptions as to instruction counts and memory references have been borne out. So far, this code is read-only from local memory; neither instructions nor temporaries need to be stored locally. The overall slowdowns due to conflicts and to Bus Coupler delay presently appear to be somewhat worse than originally estimated; however we still anticipate that the 14-processor HSMIMP will provide roughly a factor of ten increase in bandwidth over the 516 IMP.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author)	2a. REPORT SECURITY CLASSIFICATION
Bolt Beranek and Newman Inc. 50 Moulton Street Cambridge, Mass. 02138	UNCLASSIFIED
	2b. GROUP

3 REPORT TITLE

QUARTERLY TECHNICAL REPORT NO. 1

4 DESCRIPTIVE NOTES (Type of report and, inclusive dates)

5 AUTHOR(S) (First name, middle initial, last name)

Bolt Beranek and Newman Inc.

6 REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
April 1973	23	

8a. CONTRACT OR GRANT NO	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	Report No. 2541
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10 DISTRIBUTION STATEMENT

11 SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Advanced Research Projects Agency Arlington, Virginia 22209

13 ABSTRACT

The basic function of the ARPA computer network is to allow large existing computers (Hosts), with different system configurations, to communicate with each other. Each Host is connected to an Interface Message Processor (IMP), which transmits messages from its Host(s) to other Hosts and accepts messages for its Host(s) from other Hosts. There is frequently no direct communication circuit between two Hosts that wish to communicate; in these cases intermediate IMPs act as message switchers. The message switching is performed as a store and forward operation. The IMPs regularly exchange information which: allows each IMP to adapt its message routing to the conditions of its local section of the network; reports network performance and malfunctions to a Network Control Center; permits message tracing so that network operation can be studied comprehensively; allows network reconfiguration without reprogramming each IMP. The Terminal IMP (TIP), which consists of an IMP and a Multi-Line Controller (MLC), extends the network concepts by permitting the direct attachment (without an intervening Host) of up to 64 dissimilar terminal devices to the network. The Terminal IMP program provides many aspects of the Host protocols in order to allow effective communication between a terminal user and a Host process. A High Speed Modular IMP (HSMIMP) is under development; one goal of this effort is to increase IMP performance by a factor of 10.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6811

UNCLASSIFIED

Security Classification

A-31408

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Computers and Communication						
Store and Forward Communication						
ARPA Computer Network						
Interface Message Processor						
IMP						
Terminal IMP						
TIP						
Honeywell DDP-516						
Honeywell 316						
Multi-Line Controller						
MLC						
Network Control Center						
NCC						
Host Protocol						
High Speed Modular IMP						
HSMIMP						
Lockheed SUE						